

**WHAT IS CLAIMED IS:**

- 1 1. A computer implemented method comprising:  
2 detecting that a memory being managed by a virtual  
3 memory manager is constrained; and  
4 dynamically altering settings used by a sequential  
5 access read ahead process in response to the  
6 detection, wherein the altered settings are adapted to  
7 conserve memory used by the sequential access read  
8 ahead process.
- 1 2. The method of claim 1 wherein the altering further  
2 comprises:  
3 decreasing a maximum page ahead value, wherein the  
4 maximum page ahead value corresponds to a maximum  
5 number of pages read by the sequential access read  
6 ahead process.
- 1 3. The method of claim 2 further comprising:  
2 retrieving a value corresponding to a number of free  
3 page frames currently being managed by the virtual  
4 memory manager; and  
5 calculating a difference between the number of free  
6 page frames value and a minimum desired free page  
7 constant, wherein the maximum page ahead value is  
8 decreased by an amount based on the calculated  
9 difference.
- 1 4. The method of claim 2 further comprising:  
2 executing the virtual memory manager for a time  
3 interval after decreasing the maximum page ahead  
4 value;

5 determining that the memory is less constrained after  
6 the time interval has elapsed; and  
7 increasing the maximum page ahead value in response to  
8 the determination.

1 5. The method of claim 1 wherein the altering further  
2 comprises:  
3 disabling the sequential access read ahead process.

1 6. The method of claim 5 further comprising:  
2 executing the virtual memory manager for a time  
3 interval after disabling the sequential access read  
4 ahead process;  
5 determining that the memory is less constrained after  
6 the time interval has elapsed; and  
7 enabling the sequential access read ahead process in  
8 response to the determination.

1 7. A computer-implemented method of managing memory  
2 pages, wherein the memory includes a number of used  
3 pages and a number of free pages, said method  
4 comprising:  
5 retrieving a freelist value corresponding to the  
6 current number of free pages;  
7 determining that the freelist value is less than a  
8 predetermined minimum value; and  
9 dynamically altering settings used by a sequential  
10 access read ahead process in response to the  
11 detection, wherein the altered settings are adapted to

12        decrease the allocation of free pages to the  
13        sequential access read ahead process.

1    8.    The method of claim 7 wherein the altering further  
2        comprises:  
3        decreasing a current maximum page ahead value, wherein  
4        the current maximum page ahead value corresponds to a  
5        maximum number of pages read by the sequential access  
6        read ahead process.

1    9.    The method of claim 8 wherein the decreasing further  
2        comprises:  
3        calculating a shift value based on the difference  
4        between the freelist value and the predetermined  
5        minimum value; and  
6        bit shifting a constant maximum page ahead value by  
7        the shift value, the bit shifting resulting in the  
8        current maximum page ahead value.

1    10.   The method of claim 9 wherein the calculating further  
2        comprises:  
3        dividing the difference between the freelist value and  
4        the predetermined minimum value by the difference  
5        between the predetermined minimum value and a  
6        predetermined adjustment threshold, the dividing  
7        resulting in a quotient and a remainder; and  
8        setting the shift value to the quotient.

1    11.   The method of claim 8 further comprising:  
2        executing a virtual memory manager for a time interval  
3        following the decreasing, wherein the virtual memory

4       manager manages the memory pages and wherein the  
5       virtual memory manager includes the sequential access  
6       read ahead process;  
  
7       retrieving a subsequent freelist value corresponding  
8       to the number of free pages available after the time  
9       interval;  
  
10      comparing the subsequent freelist value with the  
11      current maximum page ahead value; and  
  
12      disabling the sequential access read ahead process in  
13      response to the comparison.

1    12. The method of claim 8 further comprising:  
2       executing a virtual memory manager for a time interval  
3       following the decreasing, wherein the virtual memory  
4       manager manages the memory pages and wherein the  
5       virtual memory manager includes the sequential access  
6       read ahead process;  
  
7       retrieving a subsequent freelist value corresponding  
8       to the number of free pages available after the time  
9       interval;  
  
10      determining that the subsequent freelist value is less  
11      than the predetermined minimum value;  
  
12      calculating a second shift value based on the  
13      difference between the subsequent freelist value and  
14      the predetermined minimum value; and  
  
15      bit shifting the constant maximum page ahead value by  
16      the second shift value, the bit shifting resulting in  
17      the current maximum page ahead value.

1 13. The method of claim 8 further comprising:  
2 executing a virtual memory manager for a time interval  
3 following the decreasing, wherein the virtual memory  
4 manager manages the memory pages and wherein the  
5 virtual memory manager includes the sequential access  
6 read ahead process;  
7 retrieving a subsequent freelist value corresponding  
8 to the number of free pages available after the time  
9 interval;  
10 determining that the subsequent freelist value is  
11 greater than the predetermined minimum value; and  
12 setting the current maximum read ahead value equal to  
13 the constant maximum page ahead value in response to  
14 the determination.

1 14. The method of claim 7 wherein the altering further  
2 comprises:  
3 disabling the sequential access read ahead process.

1 15. An information handling system comprising:  
2 one or more processors;  
3 a memory accessible by the processors;  
4 an operating system that controls the processors;  
5 a virtual memory manager, included with the operating  
6 system, that manages use of the memory;  
7 a nonvolatile storage area managed by the operating  
8 system and including a disk swap area used by the  
9 virtual memory manager;

10 a sequential access read ahead process performed by  
11 the operating system adapted to pre-fetch data being  
12 sequentially read from file stored in the nonvolatile  
13 storage area;

14 memory conservation software used by the virtual  
15 memory manager, the software effective to:

16 detect that a memory being managed by a virtual  
17 memory manager is constrained; and

18 dynamically alter settings used by the sequential  
19 access read ahead process in response to the  
20 detection, wherein the altered settings are  
21 adapted to conserve memory used by the sequential  
22 access read ahead process.

1 16. The information handling system of claim 15 wherein  
2 the software is further effective to:  
3 decrease a maximum page ahead value, wherein the  
4 maximum page ahead value corresponds to a maximum  
5 number of pages read by the sequential access read  
6 ahead process.

1 17. The information handling system of claim 16 wherein  
2 the software is further effective to:  
3 retrieve a value corresponding to a number of free  
4 page frames currently being managed by the virtual  
5 memory manager; and  
6 calculate a difference between the number of free page  
7 frames value and a minimum desired free page constant,  
8 wherein the maximum page ahead value is decreased by  
9 an amount based on the calculated difference.

1 18. The information handling system of claim 16 wherein  
2 the software is further effective to:  
3 execute the virtual memory manager for a time interval  
4 after decreasing the maximum page ahead value;  
5 determine that the memory is less constrained after  
6 the time interval has elapsed; and  
7 increase the maximum page ahead value in response to  
8 the determination.

1 19. The information handling system of claim 15 wherein  
2 the software is further effective to:  
3 disable the sequential access read ahead process.

1 20. The information handling system of claim 19 wherein  
2 the software is further effective to:  
3 execute the virtual memory manager for a time interval  
4 after disabling the sequential access read ahead  
5 process;  
6 determine that the memory is less constrained after  
7 the time interval has elapsed; and  
8 enable the sequential access read ahead process in  
9 response to the determination.

1 21. An information handling system comprising:  
2 one or more processors;  
3 a memory accessible by the processors;  
4 an operating system that controls the processors;

5 a virtual memory manager, included with the operating  
6 system, that manages use of the memory;

7 a nonvolatile storage area managed by the operating  
8 system and including a disk swap area used by the  
9 virtual memory manager;

10 a sequential access read ahead process performed by  
11 the operating system adapted to pre-fetch data being  
12 sequentially read from file stored in the nonvolatile  
13 storage area;

14 memory conservation software used by the virtual  
15 memory manager, the software effective to:

16 retrieve a freelist value corresponding to the  
17 current number of free pages;

18 determine that the freelist value is less than a  
19 predetermined minimum value; and

20 dynamically alter settings used by a sequential  
21 access read ahead process in response to the  
22 detection, wherein the altered settings are  
23 adapted to decrease the allocation of free pages  
24 to the sequential access read ahead process.

1 22. The information handling system of claim 21 wherein  
2 the software is further effective to:  
3 decreasing a current maximum page ahead value, wherein  
4 the current maximum page ahead value corresponds to a  
5 maximum number of pages read by the sequential access  
6 read ahead process.



1 23. The information handling system of claim 22 wherein  
2 the decreasing further comprises:  
3 calculating a shift value based on the difference  
4 between the freelist value and the predetermined  
5 minimum value; and  
6 bit shifting a constant maximum page ahead value by  
7 the shift value, the bit shifting resulting in the  
8 current maximum page ahead value.

1 24. The information handling system of claim 23 wherein  
2 the calculating further comprises:  
3 dividing the difference between the freelist value and  
4 the predetermined minimum value by the difference  
5 between the predetermined minimum value and a  
6 predetermined adjustment threshold, the dividing  
7 resulting in a quotient and a remainder; and  
8 setting the shift value to the quotient.

1 25. The information handling system of claim 22 further  
2 comprising:  
3 executing a virtual memory manager for a time interval  
4 following the decreasing, wherein the virtual memory  
5 manager manages the memory pages and wherein the  
6 virtual memory manager includes the sequential access  
7 read ahead process;  
8 retrieving a subsequent freelist value corresponding  
9 to the number of free pages available after the time  
10 interval;

11 comparing the subsequent freelist value with the  
12 current maximum page ahead value; and  
13 disabling the sequential access read ahead process in  
14 response to the comparison.

1 26. The information handling system of claim 22 further  
2 comprising:  
3 executing a virtual memory manager for a time interval  
4 following the decreasing, wherein the virtual memory  
5 manager manages the memory pages and wherein the  
6 virtual memory manager includes the sequential access  
7 read ahead process;  
8 retrieving a subsequent freelist value corresponding  
9 to the number of free pages available after the time  
10 interval;  
11 determining that the subsequent freelist value is less  
12 than the predetermined minimum value;  
13 calculating a second shift value based on the  
14 difference between the subsequent freelist value and  
15 the predetermined minimum value; and  
16 bit shifting the constant maximum page ahead value by  
17 the second shift value, the bit shifting resulting in  
18 the current maximum page ahead value.

1 27. The information handling system of claim 22 further  
2 comprising:  
3 executing a virtual memory manager for a time interval  
4 following the decreasing, wherein the virtual memory  
5 manager manages the memory pages and wherein the

6 virtual memory manager includes the sequential access  
7 read ahead process;  
8 retrieving a subsequent freelist value corresponding  
9 to the number of free pages available after the time  
10 interval;  
11 determining that the subsequent freelist value is  
12 greater than the predetermined minimum value; and  
13 setting the current maximum read ahead value equal to  
14 the constant maximum page ahead value in response to  
15 the determination.

1 28. The method of claim 27 wherein the altering further  
2 comprises:  
3 disabling the sequential access read ahead process.

1 29. A computer program product stored on a computer  
2 operable media, said computer program product  
3 comprising:  
4 means for detecting that a memory being managed by a  
5 virtual memory manager is constrained; and  
6 means for dynamically altering settings used by a  
7 sequential access read ahead process in response to  
8 the detection, wherein the altered settings are  
9 adapted to conserve memory used by the sequential  
10 access read ahead process.

1 30. The computer program product of claim 29 wherein the  
2 means for altering further comprises:  
3 means for decreasing a maximum page ahead value,  
4 wherein the maximum page ahead value corresponds to a

5 maximum number of pages read by the sequential access  
6 read ahead process.

1 31. The computer program product of claim 30 further  
2 comprising:  
3 means for retrieving a value corresponding to a number  
4 of free page frames currently being managed by the  
5 virtual memory manager; and  
6 means for calculating a difference between the number  
7 of free page frames value and a minimum desired free  
8 page constant, wherein the maximum page ahead value is  
9 decreased by an amount based on the calculated  
10 difference.

1 32. The computer program product of claim 30 further  
2 comprising:  
3 means for executing the virtual memory manager for a  
4 time interval after decreasing the maximum page ahead  
5 value;  
6 means for determining that the memory is less  
7 constrained after the time interval has elapsed; and  
8 means for increasing the maximum page ahead value in  
9 response to the determination.

1 33. The computer program product of claim 29 wherein the  
2 means for altering further comprises:  
3 means for disabling the sequential access read ahead  
4 process.

1 34. The computer program product of claim 33 further  
2 comprising:

3 means for executing the virtual memory manager for a  
4 time interval after disabling the sequential access  
5 read ahead process;

6 means for determining that the memory is less  
7 constrained after the time interval has elapsed; and

8 means for enabling the sequential access read ahead  
9 process in response to the determination.

1 35. A computer-implemented computer program product for  
2 managing memory pages, wherein the memory includes a  
3 number of used pages and a number of free pages, said  
4 computer program product comprising:  
5 means for retrieving a freelist value corresponding to  
6 the current number of free pages;  
7 means for determining that the freelist value is less  
8 than a predetermined minimum value; and  
9 means for dynamically altering settings used by a  
10 sequential access read ahead process in response to  
11 the detection, wherein the altered settings are  
12 adapted to decrease the allocation of free pages to  
13 the sequential access read ahead process.

1 36. The computer program product of claim 35 wherein the  
2 means for altering further comprises:  
3 means for decreasing a current maximum page ahead  
4 value, wherein the current maximum page ahead value  
5 corresponds to a maximum number of pages read by the  
6 sequential access read ahead process.

1 37. The computer program product of claim 36 wherein the  
2 means for decreasing further comprises:  
3 means for calculating a shift value based on the  
4 difference between the freelist value and the  
5 predetermined minimum value; and  
6 means for bit shifting a constant maximum page ahead  
7 value by the shift value, the bit shifting resulting  
8 in the current maximum page ahead value.

1 38. The computer program product of claim 37 wherein the  
2 means for calculating further comprises:  
3 means for dividing the difference between the freelist  
4 value and the predetermined minimum value by the  
5 difference between the predetermined minimum value and  
6 a predetermined adjustment threshold, the dividing  
7 resulting in a quotient and a remainder; and  
8 means for setting the shift value to the quotient.

1 39. The computer program product of claim 36 further  
2 comprising:  
3 means for executing a virtual memory manager for a  
4 time interval following the execution of the means for  
5 decreasing, wherein the virtual memory manager manages  
6 the memory pages and wherein the virtual memory  
7 manager includes the sequential access read ahead  
8 process;  
9 means for retrieving a subsequent freelist value  
10 corresponding to the number of free pages available  
11 after the time interval;

12 means for comparing the subsequent freelist value with  
13 the current maximum page ahead value; and  
14 means for disabling the sequential access read ahead  
15 process in response to the comparison.

1 40. The computer program product of claim 36 further  
2 comprising:  
3 means for executing a virtual memory manager for a  
4 time interval following the execution of the means for  
5 decreasing, wherein the virtual memory manager manages  
6 the memory pages and wherein the virtual memory  
7 manager includes the sequential access read ahead  
8 process;  
9 means for retrieving a subsequent freelist value  
10 corresponding to the number of free pages available  
11 after the time interval;  
12 means for determining that the subsequent freelist  
13 value is less than the predetermined minimum value;  
14 means for calculating a second shift value based on  
15 the difference between the subsequent freelist value  
16 and the predetermined minimum value; and  
17 means for bit shifting the constant maximum page ahead  
18 value by the second shift value, the bit shifting  
19 resulting in the current maximum page ahead value.

1 41. The computer program product of claim 36 further  
2 comprising:  
3 means for executing a virtual memory manager for a  
4 time interval following the decreasing, wherein the  
5 virtual memory manager manages the memory pages and

6        wherein the virtual memory manager includes the  
7        sequential access read ahead process;  
8        means for retrieving a subsequent freelist value  
9        corresponding to the number of free pages available  
10       after the time interval;  
11       means for determining that the subsequent freelist  
12       value is greater than the predetermined minimum value;  
13       and  
14       means for setting the current maximum read ahead value  
15       equal to the constant maximum page ahead value in  
16       response to the determination.

1    42. The computer program product of claim 35 wherein the  
2       means for altering further comprises:  
3       means for disabling the sequential access read ahead  
4       process.